Enhanced Task Modelling for Systematic Identification and Explicit Representation of Human Errors

Racim Fahssi, Célia Martinie, Philippe Palanque

▶ **To cite this version:**

Racim Fahssi, Célia Martinie, Philippe Palanque. Enhanced Task Modelling for Systematic Identification and Explicit Representation of Human Errors. 15th Human-Computer Interaction (INTERACT), Sep 2015, Bamberg, Germany. pp.192-212, 10.1007/978-3-319-22723-8_16 . hal-01387758

HAL Id: hal-01387758

https://hal.science/hal-01387758

Submitted on 26 Oct 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Open Archive TOULOUSE Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in : http://oatao.univ-toulouse.fr/
Eprints ID : 15322

The contribution was presented at INTERACT 2015 :
http://www.interact2015.org/

To cite this version : Fahssi, Racim Mehdi and Martinie De Almeida, Celia and Palanque, Philippe *Enhanced Task Modelling for Systematic Identification and Explicit Representation of Human Errors.* (2015) In: 15th IFIP TC13 Conference on Human-Computer Interaction (INTERACT 2015), 14 September 2015 - 18 September 2015 (Bamberg, Germany).

Any correspondence concerning this service should be sent to the repository administrator: staff-oatao@listes-diff.inp-toulouse.fr

Enhanced Task Modelling for Systematic Identification and Explicit Representation of Human Errors

Racim Fahssi, Célia Martinie, Philippe Palanque
Institute of Research in Informatics of Toulouse (IRIT), University Toulouse 3
118, route de Narbonne, 31062 Toulouse cedex 9, France

{fahssi, martinie, palanque}@irit.fr

Abstract. Task models produced from task analysis, are a very important element of UCD approaches as they provide support for describing users goals and users activities, allowing human factors specialists to ensure and assess the effectiveness of interactive applications. As user errors are not part of a user goal they are usually omitted from tasks descriptions. However, in the field of Human Reliability Assessment, task descriptions (including task models) are central artefacts for the analysis of human errors. Several methods (such as HET, CREAM and HERT) require task models in order to systematically analyze all the potential errors and deviations that may occur. However, during this systematic analysis, potential human errors are gathered and recorded separately and not connected to the task models. Such non integration brings issues such as completeness (i.e. ensuring that all the potential human errors have been identified) or combined errors identification (i.e. identifying deviations resulting from a combination of errors). We argue that representing human errors explicitly and systematically within task models contributes to the design and evaluation of error-tolerant interactive system. However, as demonstrated in the paper, existing task modeling notations, even those used in the methods mentioned above, do not have a sufficient expressive power to allow systematic and precise description of potential human errors. Based on the analysis of existing human error classifications, we propose several extensions to existing task modelling techniques to represent explicitly all the types of human error and to support their systematic task-based identification. These extensions are integrated within the tool-supported notation called HAMSTERS and are illustrated on a case study from the avionics domain.

1 Introduction

Task analysis and modelling approaches have always focused on the explicit representation of standard behavior of users, leaving user error analysis for later phases in the design processes [2]. This is part of the rationale underlying task analysis which is to provide an exhaustive analysis of user behavior describing goals and activities to reach these goals. Clearly, errors, mistakes and deviations are not part of the users' goals and thus left aside of tasks descriptions. This exhaustive aspect of task analysis is fundamental as it is meant to provide the basics for a global understanding of users behaviors which will serve as a basis for driving evolutions of the interactive system. However, practice (for real-life applications) shows that reaching this comprehensiveness is very hard, especially as it require a vast amount of resources. If cuts have to be made when analyzing standard activities, it is clear that infrequent or abnormal behaviors are often not considered. However, this is precisely where the emphasis should be placed in order to deal efficiently with error tolerance as error prone systems deeply impact efficiency and satisfaction. Beyond these usability-related aspects, in critical systems the cost of an operator error might put people life at stake, and this is the reason why Human Reliability Assessment (HRA) methods (such as HET, CREAM or HERT) provide means for identifying human errors. Such approaches go beyond early work of Norman on typologies of human errors [23] which have then been integrated in the action theory [24]. Indeed, they are usually associated with tasks descriptions in order to relate work and goals with erroneous behaviors of operators. However, they all exploits basic task description techniques making impossible to go beyond qualitative and quantitative temporal descriptions.

In this paper we propose the use of a detailed task description technique called HAMSTERS [21] within a HRA method to support identification of errors related to information, knowledge and devices. Beyond that, we present extensions to HAMSTERS notation in order to describe identified error within the task models. Integrating errors within a task model brings multiple advantages, the most prominent being the seamless representation of activities to reach goals and possible deviations. Such integrated representation can be exploited for building effective and error avoidant interactive systems.

The paper is structured as follows. Section 2 presents the human error domain, human reliability assessment methods and task modeling. This state of the art is used to identify limitations of current HRA methods and to identify requirements for extending task models to encompass information dedicated to user errors. Section 3 presents an extended version of the HAMSTERS notation in which genotypes and phenotypes of errors enrich "standard" task models. This section also proposes a stepwise process based on Human Error Template (HET) [36] HRA method to systematically identify user errors and to represent them in task models. Section 4 shows, on a case study, how this framework can be used and what it brings to the design and verification of error-tolerant safety critical interactive systems. Section 5 highlights benefits and limitations of the approach while section 6 concludes the paper and presents future work.

2 Related Work on Human Error and Task Modelling

Human error has received a lot of attention over the years and this section aims at presenting the main concepts related to human errors as well as the existing approaches for analyzing them. This related work section starts with the analysis of taxonomies of human errors followed by processes and methods for **identifying human errors** in socio-technical systems. Last sub-section summarizes work on **representing human errors** with a specific focus on representations based on task description.

2.1 Definition and taxonomies of human errors

Several contributions in the human factors domain deal with studying internal human processes that may lead to actions that can be perceived as erroneous from an external view point. In the 1970s, Norman, Rasmussen and Reason have proposed theoretical frameworks to analyze human error. Norman, proposed a predictive model for errors [23], where the concept of "slip" is highlighted and causes of error are rooted in improper activation of patterns of action. Rasmussen proposes a model of human performance which distinguishes three levels: skills, rules and knowledge (SRK model) [31]. This model provides support for reasoning about possible human errors and has been used to classify error types. Reason [33] takes advantages of the contributions of Norman and Rasmussen, and distinguishes three main categories of errors:

1. Skill-based errors are related to the skill level of performance in SRK. These errors can be of one of the 2 following types: a) Slip, or routine error, which is defined as a mismatch between an intention and an action [23]; b) Lapse which is defined as a memory failure that prevents from executing an intended action.
2. Rule-based mistakes are related to the rule level of performance in SRK and are defined as the application of an inappropriate rule or procedure.
3. Knowledge-based errors are related to the knowledge level in SRK and are defined as an inappropriate usage of knowledge, or a lack of knowledge or corrupted knowledge preventing from correctly executing a task.

At the same time, Reason proposed a model of human performance called GEMS [33] (Generic Error Modelling System), which is also based on the SRK model and dedicated to the representation of human error mechanisms. GEMS is a conceptual framework that embeds a detailed description of the potential causes for each error types above. These causes are related to various models of human performance. For example, a perceptual confusion error in GEMS is related to the perceptual processor of the Human Processor model [5]. GEMS is very detailed in terms of description and vocabulary (e.g. strong habit intrusion, capture errors, overshooting a stop rule …) and structuring approaches have been proposed as the Human Error Reference Table (HERT) in [25].

Causes of errors and their observation are different concepts that should be separated when analyzing user errors. To do so, Hollnagel [9] proposed a terminology based on 2 main concepts: phenotype and genotype. The phenotype of an error is defined as the erroneous action that can be observed. The genotype of the error is defined as the characteristics of the operator that may contribute to the occurrence of an erroneous action.

These concepts and the classifications above provide support for reasoning about human errors and have been widely used to develop approaches to design and evaluate interactive systems [34]. As pointed out in [26] investigating the association between a phenotype and its potential genotypes is very difficult but is an important step in order to assess the error-proneness of an interactive system. This is why most of the approaches for Human Reliability Assessment focus on this double objective, as presented in next section.

2.2 Techniques and methods for identifying Human Errors

Many techniques have been proposed for identifying which human errors may occur in a particular context and what could be their consequences in this given context. Several human reliability assessment techniques such as CREAM [10], HEART [44], and THERP [39] are based on task analysis. They provide support to assess the possibility of occurrence of human errors by structuring the analysis around task descriptions. Beyond these commonalities, THERP technique provides support for assessing the probability of occurrence of human errors. Table 1 presents an overview on the existing techniques for identifying potential human errors. For each technique, the following information is highlighted:

- Type of technique: to indicate to which scientific domain this technique is related. Values can be HEI (Human Error Identification), DC (Dependable Computing), SA (Safety Analysis) …
- Associated task modelling technique: to indicate how the user tasks are described once the task analysis has been performed. Most of them exploit HTA (Hierarchical Task Analysis notation) [1];

Table 1. Summary of techniques and methods used for identifying human errors

Name of the technique	**Type of technique**	**Task modelling**	**Tool support**	**Associated error classification (Generic/Specific)**		**Combination of errors**
Hazard and operability study (HAZOP) [15]	Safety analysis	HTA	None	Not done, Less, More, As well as, Other than, Repeated, Sooner, Later, Misordered, Part of	G	NE
Systematic human error reduction and prediction approach (SHERPA) [7]	HEI, HRA	HTA	None	Action errors, Checking errors, Communication errors, Info retrieval errors, Selection errors	S	No
Potential human error cause analysis (PHECA) [43]	HEI, HRA	HTA	None	HAZOP classification	G	NE
Cognitive reliability and error analysis method (CREAM) [10]	HEI, HRA	HTA	None	Timing, Duration, Sequence, Object, Force, Direction, Distance, Speed	S	NE
Human error assessment and reduction technique (HEART) [44]	HEI, HRA	HTA	None	None (concrete description of the human error)	S	NE
Human Error Identification In Systems Tool (HEIST) [13]	HEI, HRA	HTA	None	Skill Rule Knowledge model	S	NE
Human Error Template (HET) [36]	HEI, Human Factors	HTA	None	*Fail to execute*: Task execution incomplete, Task executed in the wrong direction, Wrong task executed, Task repeated, Task executed on the wrong interface element, *Task executed* : too early/too late/too much/too little, Misread information, other	S	No
System for Predictive Error Analysis and Reduction (SPEAR) [40]	HEI, HRA, SA	HTA	None	Action, Retrieval, Check, Selection, Transmission	G	No
Task Analysis For Error Identification (TAFEI) [2]	HEI, HF	HTA	None	Generic categories	S	NE
Technique for Human Error Assessment (THEA) [30]	HEI, HCI	HTA	None	Goals, Plans, Performing actions, Perception, Interpretation and evaluation	S	NE
Human Error Recovery and Assessment (HERA) [14]	HEI HRA	HTA	None	Omission, Timing, Sequence, Quality, Selection error, Information Transmission error, Rule Violation, Other	S	No
Tech. for Human Error Precision Rate (THERP) [39]	HRA	Not specified.	None	Omission, Commission, Selection error, Error of sequence, Time error, Qualitative error.	S	No
Tech. for the Retrospective and Predictive Analysis of Cognitive Errors in Air Traffic Control (TRACer) [35]	HEI HRA	HTA	None	Selection and Quality Timing and Sequence Communication	S	No
Task Model-Based Systematic Analysis of System Failures and Human Errors [20]	HCI, DC	HAMSTERS	HAMSTERS	HAZOP and Reason classifications	G	NE

- Tool support for task analysis and modelling: to indicate whether or not a particular Computer Aided Software Environment (CASE) tool is available to provide support for the application of the technique;
- Associated error classification: to indicate which human error classification is used to identify possible errors. ‘G’ and ‘S’ indicates whether the classification comes from a generic system failures analysis or whether it is specific to human errors;
- Capacity to deal with combination of errors: to indicate whether or not the techniques provides explicit support for identifying possible combinations of errors. Here only 2 values are possible: ‘No’ and ‘NE’ (Not Explicitly meaning that the method was not claiming explicitly that combinations of errors are handled).

For all the techniques presented above the process of identifying possible human errors highly relies on the user tasks descriptions. The task descriptions have to be precise, complete and representative of the user activities, in order to be able to identify all the possible errors. Indeed, the task description language as well as the mean to produce the description affect the quality of the analysis. However, most of them exploit Hierarchical Task Analysis (HTA) which only provides support for decomposing user goals into tasks and subtasks and for describing the sequential relationships between these tasks (in a separate textual representation called “plan”). As HTA does not provide support for describing precisely the types of user actions, the temporal ordering types that are different from a sequence of actions (such as concurrent actions, order independent actions…), as well as information and knowledge required to perform an action, errors related to these elements cannot be identified. Furthermore, as most of these techniques do not have tool support it is cumbersome to check coverage of and to store identified errors in a systematic way. For example, as HTA does not provide support for describing knowledge required to perform a task, none of these methods provide explicit support for the identification of all possible knowledge-based mistakes.

2.3 Support for representation of human errors in task model

As explained above the expressive power of the task modelling notation has a direct impact on how task models produced with these notation are likely to support the identification of errors. Many task modelling notations have been proposed over the years focusing on the representation of standard user behaviors most of the time leaving aside erroneous behaviors.

Table 2 presents a comparison of task modelling notations to assess (depending on their expressive power) their capability in identifying and representing human errors. For each notation, the following information is highlighted:

- Identification of human error: to indicate whether or not the notation provides support to systematically establish a relationship between a task model element and a component of a model of human information processing or model of human performance.
- Explicit representation of human error: to indicate whether or not the notation provides support to systematically represent human error related information in a task model.
- Explicit representation of error recovery: to indicate whether or not the notation provides support to explicitly represent recovery tasks i.e. when an error has occurred, to describe the set of actions to be performed in order to still reach the goal. While this is possible in most task modelling notations (e.g. set of action to perform after entering a wrong PIN when using a cash machine) we identify

here the fact that the notation makes explicit (or not) that this set of task is related to a user error.

Table 2. Support for describing errors and errors-related elements

Element of representation \ Task Modelling Notations		CTT [27]	COMM [12]	GOMS [4]	GTA [41]	HAMSTERS [1]	HTA [1]	SAMANTA [42]	TKS [11]
Identification of human error	Representation of refined user tasks	No	No	No	No	Yes	No	No	No
	Representation of declarative knowledge	No	No	No	No	Yes	No	Yes	Yes
	Representation of manipulated information	No	No	No	No	Yes	No	Yes	No
Explicit representation of human error	Representation of cause and observable consequence of errors (Genotype, Phenotype)	No	No	No	No	No	No	No	No
	Representation of skill based errors (Slips, Lapse)	No	No	No	No	No	No	No	No
	Representation of rule based mistakes	No	No	No	No	No	No	No	No
	Representation of knowledge based mistakes	No	No	No	No	No	No	No	No
Explicit representation of error recovery		No	No	No	No	No	No	No	No

Even though the table above demonstrates the very limited account of error handling in task modeling notation, task models have already been used to take into account possible human errors while interacting with an interactive system. Paterno and Santoro proposed a model-based technique that uses insertion of deviated human actions into task models in order to evaluate the usability of the system and to inform design [28], however, such information is presented in tables outside of the task models. This approach is relevant for human error identification but only in generic terms (as it exploits HAZOP which is a standard hazard analysis method). Palanque and Basnyat proposed a technique based on task patterns (represented in CTT) that supports human routine errors [25] description. Here a specific task model is produced in which recovery actions following errors are explicitly represented, thus ending up with two un-connected task model. Modification in one of the task model has then to be reflected in the other one increasing complexity of task modelling activities. In both contributions, no specific element of the notation are introduced thus leaving the contributions to basic task elements provided in CTT notation (and thus not covering errors related to information, knowledge … as presented above).

In order to overcome the limitations of the current task modelling notations, next section presents extensions to the HAMSTERS notation to specifically represent errors. While the extensions are made explicit on that particular task modelling technique, the underlying concepts are generic making them applicable to others.

3 Extending a Task Modelling Notation to support the identification and representation of human errors

This section presents the extensions that have been added to the HAMSTERS notation in order to provide support for systematic identification and representation of human errors in task models. We also present how this extended notation has been integrated within a human error identification technique. This process starts with an extant task model and extends it with explicit genotypes and phenotypes of errors.

3.1 HAMSTERS notation

HAMSTERS (Human – centered Assessment and Modeling to Support Task Engineering for Resilient Systems) is a tool-supported graphical task modeling notation for representing human activities in a hierarchical and structured way. At the higher abstraction level, goals can be decomposed into sub-goals, which can in turn be decomposed into activities. Output of this decomposition is a graphical tree of nodes that can be tasks or temporal operators. Tasks can be of several types (depicted in Table 3) and contain information such as a name, information details, and criticality level. Only the single user high-level task types are presented here but they can be further refined. For instance the cognitive tasks can be refined in Analysis and Decision tasks [19] and collaborative activities can be refined in several task types [17].

Table 3. Task types in HAMSTERS

	Abstract	Input	Output	I/O	Processing
Abstract	Abstract	Not Applicable	Not Applicable	Not Applicable	Not Applicable
User	User abstract	Perceptive	Motor	User	Cognitive
Interactive	Abstract interactive	Input	Output	Input/Output	Not Applicable
System	Abstract system	Output	Input	Input/Output	System

Temporal operators (depicted in Table 4 and similar to the ones in CTT) are used to represent temporal relationships between sub-goals and between activities. Tasks can also be tagged by properties to indicate whether or not they are iterative, optional or both. The HAMSTERS notation is supported by a CASE tool for edition and simulation of models. This tool has been introduced in order to provide support for task system integration at the tool level [17]. This tool supported notation also provides support for structuring a large number and complex set of tasks introducing the mechanism of subroutines [22], sub-models and components [18]. Such structuring mechanisms allow describing large and complex activities by means of task models. These structuring mechanisms enables the breakdown of a task model in several ones that can be reused in the same or different task models.

Table 4. Illustration of the operator type within hamsters

<table>
<tr><th>Operator type</th><th>Symbol</th><th>Description</th></tr>
<tr><td>Enable</td><td>T1>>T2</td><td>T2 is executed after T1</td></tr>
<tr><td>Concurrent</td><td>T1|||T2</td><td>T1 and T2 are executed at the same time</td></tr>
<tr><td>Choice</td><td>T1[]T2</td><td>T1 is executed OR T2 is executed</td></tr>
<tr><td>Disable</td><td>T1[>T2</td><td>Execution of T2 interrupts the execution of T1</td></tr>
<tr><td>Suspend-resume</td><td>T1|>T2</td><td>Execution of T2 interrupts the execution of T1, T1 execution is resumed after T2</td></tr>
<tr><td>Order Independent</td><td>T1|=|T2</td><td>T1 is executed then T2 OR T2 is executed then T1</td></tr>
</table>

HAMSTERS expressive power goes beyond most other task modeling notations particularly by providing detailed means for describing data that is required and manipulated [17] in order to accomplish tasks. Fig. 1 summarizes the notation elements to represent data. Information ("Inf:" followed by a text box) may be required for execution of a system task, but it also may be required by the user to accomplish a task. Physical objects required for performing a task can also be represented ("Phy O") as well as the device (input and/or output) with which the task is performed ("i/o D"). Declarative and situational knowledge can also be made explicit by the "SiK" and "StK" elements.

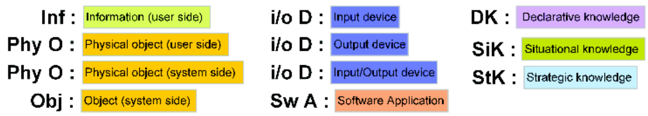


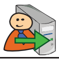
Fig. 1. Representation of Objects, Information and Knowledge with HAMSTERS Notation

3.2 HAMSTERS notation elements and relationship with genotypes

All of the above notation elements are required to be able to systematically identify and represent human errors within task models. Indeed, some genotypes (i.e. causes of human errors) can only occur with a specific type of task or with a specific element in a task model described using HAMSTERS. This relationship between classification of genotypes in human error models and task modelling elements is not trivial. For this reason, Table 5 presents the correspondences between HAMSTERS notation elements and error genotypes from the GEMS classification [32]. Such a correspondence is very useful for identifying potential genotypes on an extant task model.

It is important to note that strategic and situational knowledge elements are not present in this table. Indeed, such constructs are similar to the M (Methods) in GOMS and thus correspond to different ways of reaching a goal. As all the methods allow users to reach the goal an error cannot be made at that level and is thus not connected to a genotype.

Table 5. Correspondence between HAMSTERS elements and genotypes from GEMS [32]

Element of notation in HAMSTERS	Related genotype from GEMS [32]	
Perceptive task 	Perceptual confusion (Skill Based Error) Interference error (Skill Based Error)	
Input task  / Motor task 	Interference error (Skill Based Error) Double capture slip (Skill Based Error) Omissions following interruptions (Skill Based Error)	
Cognitive task 	Skill based errors	— Double capture slip — Omissions following interruptions — Reduced intentionality — Interference error — Over-attention errors
	Rule based mistakes	Misapplication of good rules — First exceptions — Countersigns and non-signs — Informational overload — Rule strength — General rules — Redundancy — Rigidity Application of bad rules — Encoding deficiencies — Action deficiencies
	Knowledge based mistakes	— Selectivity — Workspace limitations — Out of sight out of mind — Confirmation bias — Overconfidence — Biased reviewing — Illusory correlation — Halo effects — Problems with causality — Problems with complexity
Information **Inf :** Information	Double capture slip, Omissions following interruptions, Interference error, all of the Rule Based Mistakes and Knowledge Based Mistakes	
Declarative knowledge **DK :** Declarative	All of the Knowledge Based Mistakes	

3.3 Extensions to HAMSTERS to describe user errors

Several notation elements have been added to HAMSTERS in order to allow explicit representation of both genotypes and phenotypes of errors. Table 6 summarizes these notation elements that can be used to describe an observable consequence of an error (phenotype) and its potential associated causes (genotypes).
In that table the first column lists the types of errors following GEMS classification. The second column makes the connection with the SRK classification as previously performed in [32]. Third column present the new notation elements in HAMSTERS for describing genotypes of errors as well as how they relate to the classifications on human error. Four new elements are added: Slips, Lapses, Rule-Based Mistakes and Knowledge-Based Mistakes. As for phenotypes only one notation element is proposed. Indeed, the phenotype (i.e. how the errors is made visible) only need to be explicitly represented, the label beneath it providing a textual description while its relationship to

the causes is made by connecting genotypes to it. Such connections will be presented in details in the case study section.

Table 6. Representation of genotypes and phenotypes in HAMSTERS

Type of error (GEMS [32])	Level of Performance from [31]	Representation of genotype in HAMSTERS	Representation of phenotype in HAMSTERS
Slip	Skill-based		
Lapse			
Mistake	Rule-based		
	Knowledge-based		

3.4 Modelling process

In this section, we show how we have integrated HAMSTERS extended notation with the HET [36] technique. HAMSTERS could be used to replace HTA in any other human error identification method based on task description, but we have chosen HET because it provides a detailed process and because it has been demonstrated in [36] to be more accurate than other techniques such as SHERPA and HAZOP [37].

Fig. **2** presents a modified version of the HET process and provides support for identifying genotypes and phenotypes of possible human errors by embedding error descriptions in the task models that have been produced to describe user activities. The extended process starts with a task analysis and description phase (as for the original HET one), but in our case the produced task models are refined to represent perceptive, cognitive and motor user tasks as well as information and knowledge required to perform the tasks. These models take full advantage of the expressive power of HAMSTERS that has been presented in section 3. All the modifications made with respect to the original process have been made explicit by using various shades of grey.

Next step in the process exploits the task type–genotypes correspondence table (Table 5), to provide support for systematic identification of genotypes associated to perceptive, cognitive, motor and interactive input tasks, but also to the related phenotypes. The likelihood and criticality of a genotype are inserted as properties of the instance of represented genotype. This is performed in HAMSTERS tool by specific properties associated to the genotypes icons. Similarly, likelihood and criticality of a phenotype can also be described using properties of the instance of a represented phenotype. Likelihood of a phenotype may be a combination of likelihood of related genotypes. Once all of the possible genotypes and phenotypes have been identified and described in the task model, the human error identification and representation technique is applied to the next task model. Once all of the models have been analyzed, a last step is performed (see bottom left activity in Fig. 2) in order to determine, for each task model that embeds human error descriptions, which phenotypes may be propagated to other task models. Several phenotypes may be associated to an observable task, but not all of them may happen in a particular scenario.

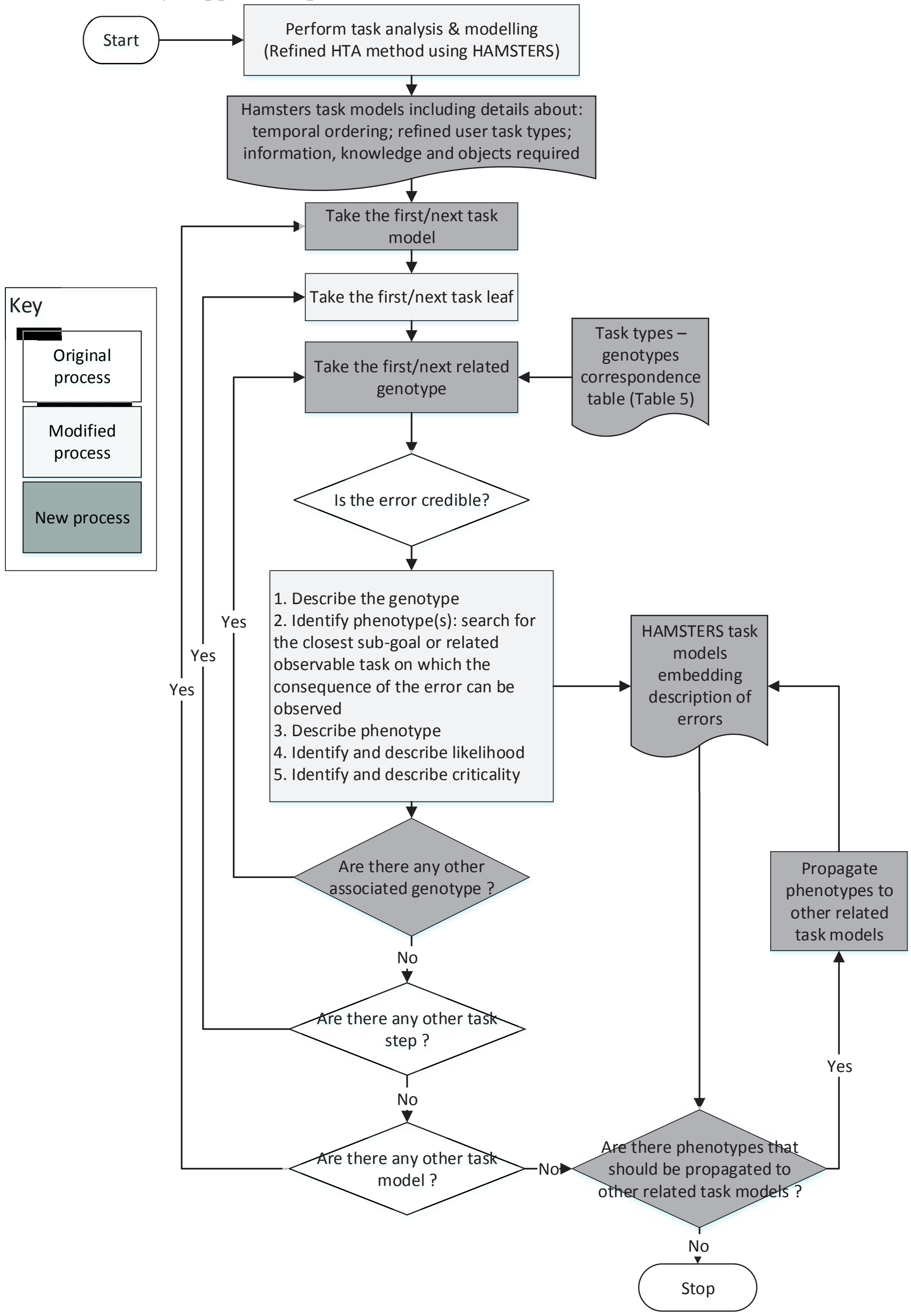


Fig. 2. Human error identification and description process extended from HET [36]

4 Illustrative example from an avionics case study

This section presents an excerpt of task models produced by the application of the process presented above for identification and representation to a case study. The case study belongs to the aeronautics domain and more precisely deals with pilot tasks exploiting a weather radar cockpit application. This section aims at illustrating how the HAMSTERS extensions can be applied to human operations on a real-life application. Due to space constraints, the application of all new elements of notation are not shown in this article but most of them are.

4.1 Presentation of the weather radar case study

Weather radar (WXR) is an application currently deployed in many cockpits of commercial aircrafts. It provides support to pilots' activities by increasing their awareness of meteorological phenomena during the flight journey, allowing them to determine if they may have to request a trajectory change, in order to avoid adverse weather conditions such as storms or precipitations. In this case study, we particularly focus on the tasks that have to be performed by a pilot to check the weather conditions on the current flight path.

a) 

b) 

Fig. 3. Image of a) the numeric part of weather radar control panel b) physical manipulation of the range of the weather radar

Fig. 3 presents a screenshot of the weather radar control panels, used to operate the weather radar application. These panels provides two functionalities to the crew. The first one is dedicated to the mode selection of weather radar and provides information about status of the radar, in order to ensure that the weather radar can be set up correctly. The operation of changing from one mode to another can be performed in the upper part of the panel (mode selection section).

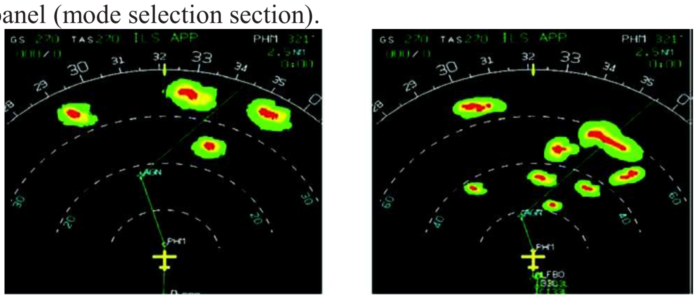


Fig. 4. Screenshots of weather radar displays

The second functionality, available in the lower part of the window, is dedicated to the adjustment of the weather radar orientation (Tilt angle). This can be done in an automatic way or manually (Auto/manual buttons). Additionally, a stabilization function aims to keep the radar beam stable even in case of turbulences. The right-hand part of Fig. 3 (labelled "b)") presents an image of the controls used to configure radar display, particularly to set up the range scale (right-hand side knob with ranges 20, 40, … nautical miles).

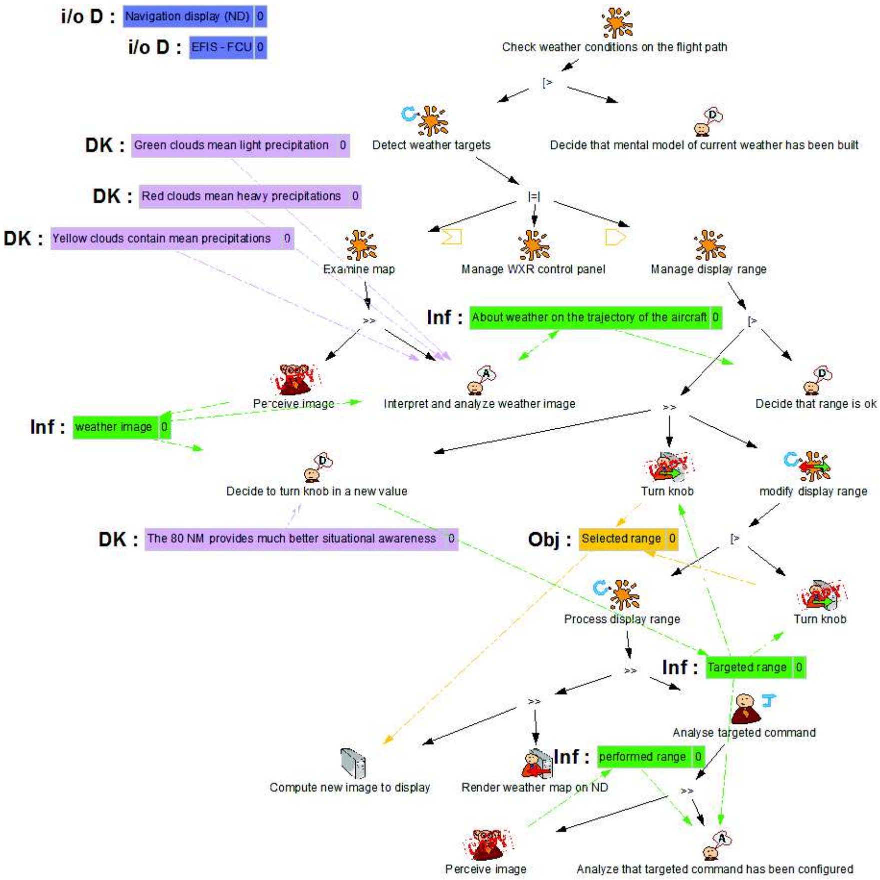


Fig. 5. Task model of the "Check weather conditions on the flight path" task

Fig. 4 shows screenshots of weather radar displays according to two different range scales (40 NM for the left display and 80 NM for the right display). Spots in the middle of the images show the current position, importance and size of the clouds. Depending on the color of the clouds in the navigation display (Fig. 4), pilots can determine whether or not the content of the clouds is dangerous for the aircraft. For example, the red color highlights the fact that the clouds contain heavy precipitations. Such information is needed in order to ensure that the current or targeted flight plan are safe.

4.2 Task model of the task "Check weather conditions on the flight path"

Fig. 5 presents the description, with HAMSTERS elements of notation, of the activities that have to be performed to check the weather conditions on the flight path.

The tasks presented in this model describe how the pilot builds a mental model of the current weather from information gathered on the navigation display (Fig. 4). For a pilot, checking weather conditions is very important as it provides support for deciding to maintain or change the current trajectory of the aircraft. This task is decomposed into 3 sub tasks:

- “Examine Map”: the pilot perceives and examines the radar image of the weather, which is displayed on the navigation display (see Fig. 4). To perform this analysis, the pilot has to know the meaning of the weather representations (described with declarative knowledge notation elements in Fig. 5 such as "Green light clouds mean precipitation").
- “Manage WXR control panel”: This sub task is represented by a subroutine, and linked to another task model, which describes the tasks that have to be performed to control the WXR modes.
- “Manage Display Range”: This sub task describes the actions that have to be performed by the pilot in order to change the range of the WXR display with using the physical knob “range” (illustrated in Fig. 3 b)). The pilot has to turn the knob to modify the range, and then to wait for the radar image to be refreshed on the navigation display (Fig. 4).

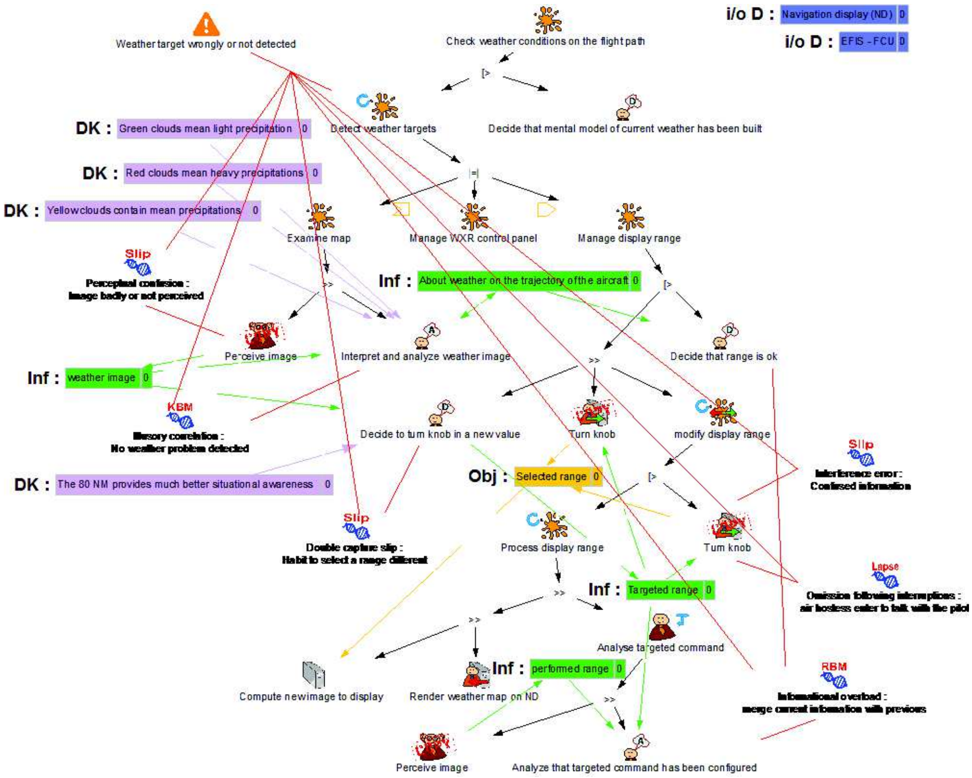


Fig. 6. Task model of the “check weather conditions on the flight path” task embedding the description of potential errors

4.3 Task model with human errors

Fig. 6 presents a modified version of the "Check weather conditions on the flight path" task model. This new version embeds the descriptions of possible human errors (genotype and phenotypes) which have been identified while applying the human error identification process.
Each human task and interactive input task is connected to one (or several) genotype(s), indicating possible cause(s) of errors. Genotypes are then connected to phenotypes, which are the observable consequences of the errors. For example, the "Perceive image" perception task is connected to the genotype "Perceptual confusion: image badly or not perceived" (zoomed in view in Fig 7). This genotype is also connected to the phenotype "Weather target wrongly or not detected". In the same way, the "Interpret and analyze" cognitive analysis task, which requires particular knowledge to be performed (the "DK" labeled rectangles containing declarative knowledge about relationships between the color of visual artefacts in the navigation display and the composition of the clouds) is connected to the knowledge based mistake "Illusory correlation: No weather problem detected". This means that a wrong user knowledge association could cause a non-detection of a weather issue on the flight path. And this genotype is also connected to the phenotype "Weather target wrongly or not detected".

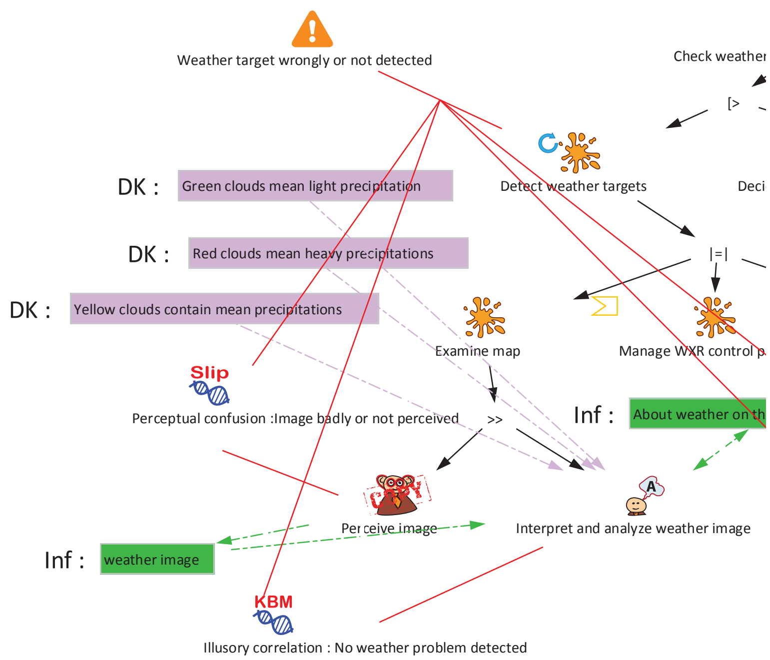


Fig 7. "Examine map" sub-task of the "check weather conditions on the flight path" task embedding the description of potential errors

5 Benefits and limitations of the approach

The stepwise refinement process of task models presented in section 3.4 and its application to the case study in section 4 have demonstrated the possibility to exploit the extended version of HAMSTERS to support identification and description of operator errors on an existing task model.

While this is critical in order to identify parts in a system that might be error prone or parts in the system that are not tolerant to operators errors it is also true that the task models enriched with error artefacts are gathering a lot of information that might decrease their understandability and modifiability. We currently favor the expressiveness of the notation and of the resulting task models than legibility and understandability. These two aspects are currently being addressed at tool level providing multiple filtering mechanisms for hiding (in a temporary way) information that the analyst is not focusing on. For instance, all the information elements can be hidden, as the genotypes and the phenotypes if the current activity is to focus on sequencing of tasks.

The main objective of the approach is to support redesign activities when error prone designs have been identified. Such redesign would take place through an iterative design process involving co-evolution of tasks and systems as presented in [3] but development costs are clearly increased. This is the reason why such an approach would be also useful for supporting certification activities in critical systems. For instance, as stated in [6] CS25-1302 annex E 1-F-1, “Flight deck controls must be installed to allow accomplishment of these tasks and information necessary to accomplish these tasks must be provided ” and in CS 25-1309 “stems and controls, including indications and annunciations must be designed to minimize crew errors, which could create additional hazards”. This CS 25 document consists in a list of requirement that have to be fulfilled in order for aircraft manufacturers to go successfully through certification processes (which are managed by regulatory authorities and/or third parties). The two highlighted requirements demonstrate that certification can only be successful using a complete and unambiguous description of operator’s tasks and by ensuring that equipment (called system in this paper) are not error prone.

Finally, it is important to note that the process proposed and its associated tool-supported notation remain a manual expert-based activity. This is made clearly visible by the “is the error credible?” step in the process where identification of errors can only come from deep understanding of operators activities and possible deviations.

6 Conclusion

In this paper we have presented a way of taking into account in a systematic way abnormal user behavior by extending previous work in the area of task modelling and human error analysis and identification.

We proposed the use of several classifications in human error and integrated them into an analysis and modelling process exploiting new extensions in the task modelling notation HAMSTERS. These extensions make it possible to explicitly represent genotypes and phenotypes of operator errors and to describe their relationships.

These contributions have been applied to a real-life case study in the field of aeronautics demonstrating most of the aspects of the contributions. However, errors related to strategic knowledge and errors related to temporal ordering (e.g. the task model describes a sequence of tasks but the operator performs them in parallel) were not presented even though covered by the approach.

As identified in “Benefits and Limitations” section, this work targets at supporting certification activities for critical systems and more precisely cockpits of large aircrafts. However, thanks to the tool support provided by HAMSTERS (which make human error identification and description less resource consuming) the approach is also applicable to other domains where errors are damaging, in terms of human life, economics, prestige, trust …

References

1. Annett, J. (2003). Hierarchical task analysis. Handbook of cognitive task design, 17-35.
2. Baber, C., & Stanton, N. A. (1994). Task analysis for error identification: a methodology for designing error-tolerant consumer products. Ergonomics, 37(11), 1923-1941.
3. Barboni E., Ladry J-F., Navarre D., Palanque P., Winckler M: Beyond modelling: an integrated environment supporting co-execution of tasks and systems models. ACM SIGCHI conference Engineering Interactive Computing Systems (EICS 2010), 165-174 ACM DL
4. Card, S. K., Newell, A., & Moran, T. P. (1983). The psychology of human-computer interaction.
5. Card, S., Moran, T., Newell, A. The model human processor: An engineering model of human performance. John Wiley & Sons, 1986.
6. EASA CS 25: Certification for large Aeroplanes. 2007. http://www.easa.europa.eu/agency-measures/certification-specifications.php#CS-25
7. Embrey, D E. SHERPA: a systematic human error reduction and prediction approach. Int. Topical Meeting on Advances in Human Factors in Nuclear Power Systems, (1986)
8. Forbrig, P., Martinie, C., Palanque, P., Winckler, M., Fahssi,R. Rapid Task-Models Development Using Sub-models, Sub-routines and Generic Components. Proc.of HCSE 2014, pp. 144-163.
9. Hollnagel E. “The phenotype of erroneous actions Implications for HCI design,” in G. R. S. Weir and J. L. Alty, Eds , Human Computer Interaction and the Complex Systems, London: Academic Press, 1991.
10. Hollnagel, Erik. Cognitive reliability and error analysis method (CREAM). Elsevier, 1998
11. Johnson, H., & Johnson, P. (1991). Task Knowledge Structures: Psychological basis and integration into system design. Acta psychologica, 78(1), 3-26.
12. Jourde, F., Laurillau, Y., & Nigay, L. (2010, June). COMM notation for specifying collaborative and multimodal interactive systems. In Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems (pp. 125-134). ACM.
13. Kirwan, B. (1994). A guide to practical human reliability assessment. CRC Press.
14. Kirwan, B. (1996). Human Error Recovery and Assessment (HERA) Guide. Project IMC/GNSR/HF/5011, Industrial Ergonomics Group, School of Manufacturing Engineering, University of Birmingham, March.
15. Kletz, T. HAZOP and HAZAN - notes on the identification and assessment of hazards. Institute of Chemical Engineers, Rugby (1974)
16. Kontogiannis, T., Malakis, S. A proactive approach to human error detection and identification in aviation and air traffic control. Safety Science, 47(5), 693e706. (2009).
17. Martinie, C., Barboni, E., Navarre, D., Palanque, P., Fahssi, R., Poupart, E., Cubero-Castan, E. Multi-models-based engineering of collaborative systems: application to collision avoidance operations for spacecraft. Proc. of EICS 2014, pp. 85-94.
18. Martinie, C., Palanque, P., & Winckler, M. (2011). Structuring and composition mechanisms to address scalability issues in task models. In Human-Computer Interaction–INTERACT 2011 (pp. 589-609). Springer Berlin Heidelberg.
19. Martinie, C., Palanque, P., Barboni, E., & Ragosta, M. (2011, October). Task-model based assessment of automation levels: application to space ground segments. In Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on (pp. 3267-3273). IEEE.
20. Martinie, C., Palanque, P., Fahssi, R., Blanquart, J-P., Fayollas, C., Seguin, C. Task Models Based Systematic Analysis of both System Failures and Human Errors. IEEE Transactions on Human Machine Systems, special issue on Systematic Approaches to Human-Machine Interface: Improving Resilience, Robustness, and Stability, to appear.
21. Martinie, C., Palanque, P., Ragosta, M., & Fahssi, R. (2013, August). Extending procedural task models by systematic explicit integration of objects, knowledge and information. In Proceedings of the 31st European Conference on Cognitive Ergonomics (p. 23). ACM.

22. Martinie, C., Palanque, P., Winckler, M. Structuring and Composition Mechanism to Address Scalability Issues in Task Models. IFIP TC13 Conference on Human-Computer Interaction (INTERACT 2011), Springer Verlag, LNCS 6948, pp. 589--609 (2011).
23. Norman, D. A. (1981). Categorization of action slips. Psychological review, 88(1), 1.
24. Norman, D. A. The Psychology of Everyday Things. New York: Basic Book, 1988.
25. Palanque, P., & Basnyat, S., Task Patterns for Taking Into Account in an Efficient and Systematic Way Both Standard and Erroneous User Behaviours, 2004, Proc. of Int. Conf. on Human Error, Safety and System Development (HESSD), pp. 109-130.
26. Papatzanis, G, Curzon, P., Blandford, A. Identifying Phenotypes and Genotypes: A Case Study Evaluating an In-Car Navigation System. EHCI/DS-VIS 2007, pp. 227-242.
27. Paternò, F., Mancini, C., & Meniconi, S. (1997, January). ConcurTaskTrees: A diagrammatic notation for specifying task models. In Human-Computer Interaction INTERACT'97 (pp. 362-369). Springer US.
28. Paterno, F., Santoro, C., "Preventing user errors by systematic analysis of deviations from the system task model", 2002, Int. Journal on Human Computing Systems, Elsevier, vol. 56, n. 2,pp. 225-245.
29. Phipps, D. L., Meakin, G. H., & Beatty, P. C. (2011). Extending hierarchical task analysis to identify cognitive demands and information design requirements. Applied ergonomics, 42(5), 741-748.
30. Pocock, S., Harrison, M., Wright, P., & Johnson, P. (2001, July). THEA: a technique for human error assessment early in design. In Human-Computer Interaction: INTERACT (Vol. 1, pp. 247-254).
31. Rasmussen, J. Skills, rules, knowledge: signals, signs and symbols and other distinctions in human performance models, IEEE transactions: Systems, Man &Cybernetics, 1983.
32. Reason J., "*Human Error*", 1990, Cambridge University Press.
33. Reason, J T. Generic error modelling system: a cognitive framework for locating common human error forms. New technology and human error, 63, 86. 1987.
34. Rizzo, A., Ferrante, D., & Bagnara, S. (1995). Handling human error. Expertise and technology: Cognition & human-computer cooperation, 195-212.
35. Shorrock, S. T., & Kirwan, B. (2002). Development and application of a human error identification tool for air traffic control. Applied Ergonomics, 33(4), 319-336.
36. Stanton, N. A., Harris, D., Salmon, P. M., Demagalski, J. M., Marshall, A., Young, M. S. & Waldmann, T. (2006). Predicting design induced pilot error using HET (human error template)-A new formal human error identification method for flight decks.
37. Stanton, N. A., Salmon, P. M., Harris, D., Marshall, A., Demagalski, J., Young, M. S. & Dekker, S. (2008). Predicting pilot error on the flight deck: Validation of a new methodology and a multiple methods and analysis approach to enhancing error prediction sensitivity.
38. Swain, A D and Guttmann, H E. A handbook of human reliability analysis with emphasis on nuclear power plant applications. USNRC-Nureg/CR-1278, DC 20555 (1983)
39. Swain, A. D., Guttman, H. E. Handbook of Human Reliability Analysis with Emphasis on Nuclear Power Plant Applications, Final report. NUREG/CR- 1278. SAND80-0200. RX, AN. US Nuclear Regulatory Commission, August 1983.
40. The Center for Chemical Process Safety. (1994). Guidelines for preventing human error in process safety. New York: American Institute of Chemical Engineers.
41. Van Der Veer, G. C., Lenting, B. F., & Bergevoet, B. A. (1996). GTA: Groupware task analysis—Modeling complexity. Acta Psychologica, 91(3), 297-322.
42. Villaren, T., Coppin, G., & Leal, A. (2012). Modeling task transitions to help designing for better situation awareness. ACM SIGCHI EICS conference p. 195-204.
43. Whailey, S P 'Minimising the cause of human error' in Libberton, G P (ed) Proc lOth Advances in Reliability Technology Symposium Elsevier, London (1988)
44. Williams, J. C. A data-based method for assessing and reducing human error to improve operational performance. In Human Factors and Power Plants, IEEE, 1988. p. 436-450.
45. Zapf, D., & Reason, J.T. (1994). Introduction: Human errors and error handling. Applied Psychology: An International Review, 43(4), 427-432.